JAVA.UTIL.SCANNER CLASS

http://www.tutorialspoint.com/java/util/java_util_scanner.htm

Introduction

The **java.util.Scanner** class is a simple text scanner which can parse primitive types and strings using regular expressions.Following are the important points about Scanner:

- A Scanner breaks its input into tokens using a delimiter pattern, which by default matches whitespace.
- A scanning operation may block waiting for input.
- A Scanner is not safe for multithreaded use without external synchronization.

Class declaration

Following is the declaration for **java.util.Scanner** class:

```
public final class Scanner
   extends Object
      implements Iterator<String>
```

Class constructors

S.N.	Constructor & Description
1	**Scanner(File source)** This constructs a new Scanner that produces values scanned from the specified file.
2	**Scanner(File source, String charsetName)** This constructs a new Scanner that produces values scanned from the specified file.
3	**Scanner(InputStream source)** This constructs a new Scanner that produces values scanned from the specified input stream..
4	**Scanner(InputStream source, String charsetName)** This constructs a new Scanner that produces values scanned from the specified input stream.
5	**Scanner(Readable source)** This constructs a new Scanner that produces values scanned from the specified source.
6	**Scanner(ReadableByteChannel source)** This constructs a new Scanner that produces values scanned from the specified channel.
7	**Scanner(ReadableByteChannel source, String charsetName)** This constructs a new Scanner that produces values scanned from the specified channel.
8	**Scanner(String source)** This constructs a new Scanner that produces values scanned from the specified string.

Class methods

S.N.	Method & Description
1	void close() This method closes this scanner.
2	Pattern delimiter() This method returns the Pattern this Scanner is currently using to match delimiters.
3	String findInLine(Pattern pattern) This method attempts to find the next occurrence of the specified pattern ignoring delimiters.
4	String findInLine(String pattern) This method attempts to find the next occurrence of a pattern constructed from the specified string, ignoring delimiters.
5	String findWithinHorizon(Pattern pattern, int horizon) This method attempts to find the next occurrence of the specified pattern.
6	String findWithinHorizon(String pattern, int horizon) This method attempts to find the next occurrence of a pattern constructed from the specified string, ignoring delimiters.
7	boolean hasNext() This method returns true if this scanner has another token in its input.
8	boolean hasNext(Pattern pattern) This method returns true if the next complete token matches the specified pattern.
9	boolean hasNext(String pattern) This method returns true if the next token matches the pattern constructed from the specified string..
10	boolean hasNextBigDecimal() This method returns true if the next token in this scanner's input can be interpreted as a BigDecimal using the nextBigDecimal() method.
11	boolean hasNextBigInteger() This method returns true if the next token in this scanner's input can be interpreted as a BigInteger in the default radix using the nextBigInteger() method.
12	boolean hasNextBigInteger(int radix) This method returns true if the next token in this scanner's input can be interpreted as a BigInteger in the specified radix using the nextBigInteger() method.
13	boolean hasNextBoolean() This method returns true if the next token in this scanner's input can be interpreted as a boolean value using a case insensitive pattern created from the string "true|false".
14	boolean hasNextByte() This method returns true if the next token in this scanner's input can be interpreted as a byte value in the default radix using the nextByte() method..
15	boolean hasNextByte(int radix) This method returns true if the next token in this scanner's input can be interpreted as a byte value in the specified radix using the nextByte() method..

16	boolean hasNextDouble() This method returns true if the next token in this scanner's input can be interpreted as a double value using the nextDouble() method.
17	boolean hasNextFloat() This method Returns true if the next token in this scanner's input can be interpreted as a float value using the nextFloat() method.
18	boolean hasNextInt() This method returns true if the next token in this scanner's input can be interpreted as an int value in the default radix using the nextInt() method.
19	boolean hasNextInt(int radix) This method returns true if the next token in this scanner's input can be interpreted as an int value in the specified radix using the nextInt() method.
20	boolean hasNextLine() This method returns true if there is another line in the input of this scanner.
21	boolean hasNextLong() This method returns true if the next token in this scanner's input can be interpreted as a long value in the default radix using the nextLong() method.
22	boolean hasNextLong(int radix) This method returns true if the next token in this scanner's input can be interpreted as a long value in the specified radix using the nextLong() method.
23	boolean hasNextShort() This method returns true if the next token in this scanner's input can be interpreted as a short value in the default radix using the nextShort() method.
24	boolean hasNextShort(int radix) This method returns true if the next token in this scanner's input can be interpreted as a short value in the specified radix using the nextShort() method.
25	IOException ioException() This method returns the IOException last thrown by this Scanner's underlying Readable.
26	Locale locale() This method returns this scanner's locale.
27	MatchResult match() This method returns the match result of the last scanning operation performed by this scanner.
28	String next() This method finds and returns the next complete token from this scanner..
29	String next(Pattern pattern) This method returns the next token if it matches the specified pattern..
30	String next(String pattern) This method returns the next token if it matches the pattern constructed from the specified string..
31	BigDecimal nextBigDecimal() This method scans the next token of the input as a BigDecimal.
32	BigInteger nextBigInteger()

	This method Scans the next token of the input as a BigInteger.
33	BigInteger nextBigInteger(int radix) This method scans the next token of the input as a BigInteger.
34	boolean nextBoolean() This method scans the next token of the input into a boolean value and returns that value.
35	byte nextByte() This method scans the next token of the input as a byte.
36	byte nextByte(int radix) This method scans the next token of the input as a byte.
37	double nextDouble() This method scans the next token of the input as a double.
38	float nextFloat() This method scans the next token of the input as a float.
39	int nextInt() This method scans the next token of the input as an int.
40	int nextInt(int radix) This method scans the next token of the input as an int.
41	String nextLine() This method advances this scanner past the current line and returns the input that was skipped.
42	long nextLong() This method scans the next token of the input as a long.
43	long nextLong(int radix) This method scans the next token of the input as a long.
44	short nextShort() This method scans the next token of the input as a short.
45	short nextShort(int radix) This method scans the next token of the input as a short.
46	int radix() This method returns this scanner's default radix..
47	void remove() The remove operation is not supported by this implementation of Iterator.
48	Scanner reset() This method resets this scanner.
49	Scanner skip(Pattern pattern) This method skips input that matches the specified pattern, ignoring delimiters.
50	Scanner skip(String pattern) This method skips input that matches a pattern constructed from the specified string.
51	String toString()

	This method returns the string representation of this Scanner.
52	Scanner useDelimiter(Pattern pattern) This method sets this scanner's delimiting pattern to the specified pattern.
53	Scanner useDelimiter(String pattern) This method sets this scanner's delimiting pattern to a pattern constructed from the specified String.
54	Scanner useLocale(Locale locale) This method sets this scanner's locale to the specified locale.
55	Scanner useRadix(int radix) This method Sets this scanner's default radix to the specified radix.

Methods inherited

This class inherits methods from the following classes:

- java.util.Object